

# **Form Master 2008**

## User Manual

Created: Friday, September 04, 2009

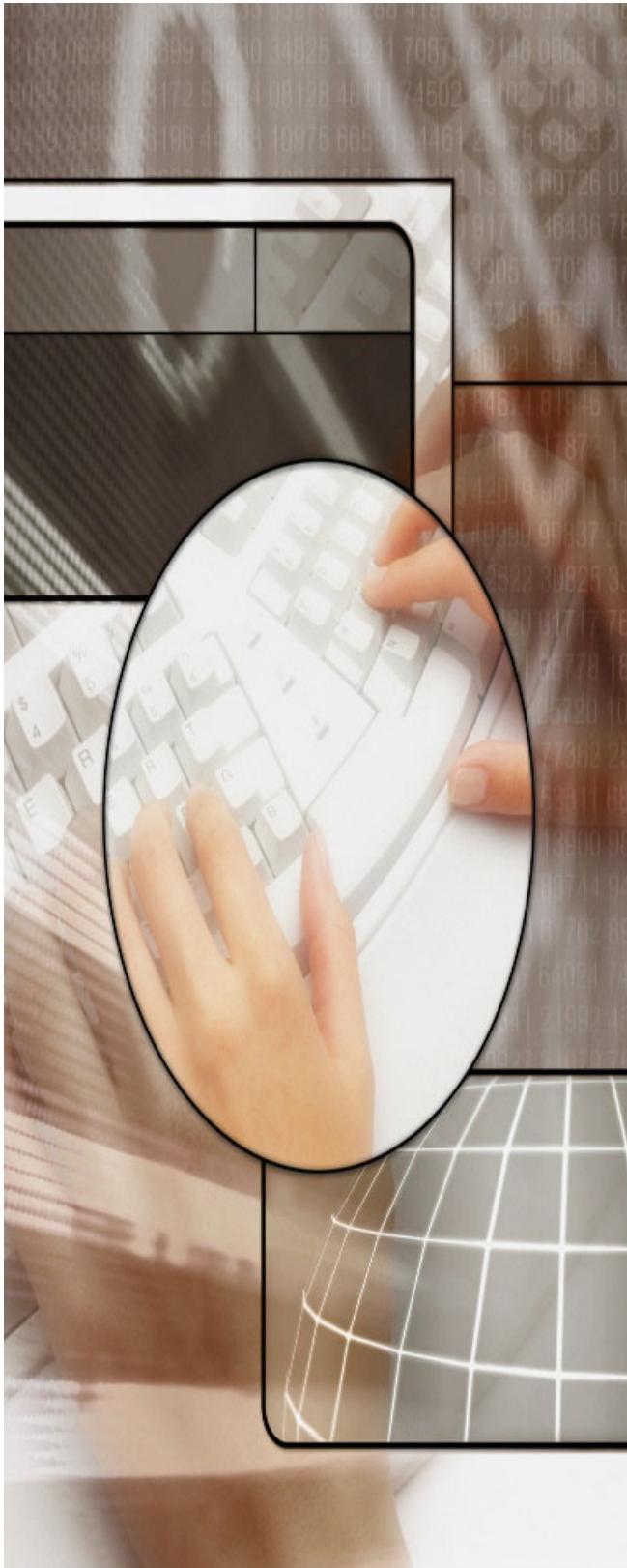




## Form Master 2008

### Welcome

Welcome to the Form Master 2008 User Guide. The guide presented here will help you the user, get up and running quickly with Form Master 2008 for DotNetNuke®





# Table of Contents

<b>1</b>	<b>Welcome</b>	<b>1</b>
	Support	2
<b>2</b>	<b>Installation</b>	<b>3</b>
	System Requirements	4
	Installing Form Master 2008	5
<b>3</b>	<b>Getting Started</b>	<b>6</b>
	About Form Master	7
	Form Submit Workflow	8
	Form Elements	9
	Menu Items	10
<b>4</b>	<b>Settings</b>	<b>11</b>
	Form Settings	12
	Other Settings	14
<b>5</b>	<b>Basic Functions</b>	<b>18</b>
	Form Layout	19
	Tab Index	21
	Adding Fields	22
	Common Options	23
	Textbox	25
	Checkbox	26
	Dropdown List	27
	Listbox	28
	Checkbox List	29
	Radiobutton List	30
	Label	31
	File Upload	32
	Submit Button	33
	Hidden Field	34
	Field Options	36
	Auto Fill Options	37
	Textbox Options	38
	Field Query Options	39
	Send Value By Options	40
	Validation	41
	Required Field Validator	42
	Regular Expression Validator	43
	Compare Validator	44
	Checkbox List Validator	45
	Validation Summary	46
	Localization	47
	File Management	48
<b>6</b>	<b>Data Manager</b>	<b>49</b>
	The Connection String	50
	Map Existing Table	51
	Table From Form	52

Form From Table	53
The Mapped Form	54
Edit Mappings	56
External Table Mask	57
<b>7    Formatting</b>	<b>58</b>
Form Formatting	59
Email Formatting	61
XSL Templates	62
Tokenized Templates	63
Custom Template	64
<b>8    Reports</b>	<b>65</b>
Viewing Reports	66
Downloading Reports	67
<b>9    Messaging</b>	<b>68</b>
Form Heading	69
Form Thank You	70
Auto Responder	71
<b>10   Advanced Functions</b>	<b>72</b>
Silent Post	73
Field Rules	74
User Manager	76

Form Master 2008

## User Guide



Installation



Getting Started



Settings



Basic Functions



Data Manager



Formatting



Reports



Messaging



Advanced Functions

### Welcome

Welcome to the Form Master 2008 online help system. Browse through the help pages by clicking on the links above or selecting pages in the table of contents. To quickly find specific product information, enter search criteria in the search box above and click the search button.

### Ask Someone

If you're unable to find what you're looking for in this help system, try these alternative [resources](#), or contact our [customer support department](#).

### Most popular pages

- [Welcome](#)
- [Support](#)
- [Settings](#)
- [System Requirements](#)
- [Other Settings](#)
- [Getting Started](#)
- [Installation](#)
- [Validation](#)
- [Data Manager](#)
- [About Form Master](#)

[Home](#) > [Welcome](#) > [Support](#)

Get help from Customer Support or other resources

### Ask Someone

Post a question or search for an answer in our online [User Forum](#).

### Knowledgebase

Search the [Knowledgebase](#) for common bugs, tips & tricks, and compatibility issues with other software.

### Customer Support

Contact our customer support department by email at [Email Support](#)



If you're unable to find what you're looking for in this help system, try these alternative resources:

- [User Forum](#)
- [Knowledgebase](#)
- [FAQ](#)

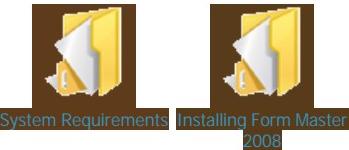
Contact our support department:  
Email: [Email Support](#)

---

Home > Installation

## Installation

Topics in this section



See also

Welcome  
Getting Started  
Settings  
Basic Functions  
Data Manager  
Formatting  
Reports  
Messaging  
Advanced Functions

[Home](#) > [Installation](#) > [System Requirements](#)

## System Requirements

### Platform:

DotNetNuke® Version 4.5.2 or Later. Form Master 2008 has been tested to DotNetNuke® Version 5.1.2 which was the latest version available when this help was published.

### Database:

Form Master 2008 is designed to use the Microsoft® SQL Server version listed below.

- SQL Server 2005 Express
- SQL Server 2005 Standard, Workgroup, and Enterprise
- SQL Server 2008 Express
- SQL Server 2008 Standard, Workgroup, and Enterprise

### See also

[Installing Form Master 2008](#)

---

[Home](#) > [Installation](#) > [Installing Form Master 2008](#)

## Installing Form Master 2008

BACKUP YOUR DATABASE!!!

BACKUP YOUR FILES!!!

BEFORE INSTALLING ANY MODULE!!!

\*\*\*\*\* IMPORTANT \*\*\*\*\*  
If you have modified the default Tokenized or XSL templates, please  
backup these files prior to installing the module, then replace them  
after installation is complete.  
\*\*\*\*\*

1. Unzip the file FormMaster\_DNN4\_02.XX.XX\_UNZIP\_ME.zip or FormMaster\_DNN4\_02.XX.XX\_SRC.zip
2. Included in the file is FormMaster\_DNN4\_02.XX.XX\_PA.zip
3. Login as host
4. Under the host menu select Module Definitions
5. Hover over the Module Definitions Action Menu, select Upload New Module
6. Click the Browse button and find FormMaster\_DNN4\_02.XX.XX\_PA.zip
7. Click Add
8. After the file is done uploading click Upload New File
9. On the results page check that there were no errors installing the module
10. You are done

As always it is recommended that you install into a test environment  
similar to your production environment, and of course backup all  
data and disk files before installing or upgrading any module in test  
or production.

DO NOT DELETE ANY PREVIOUS VERSION OF FORM MASTER  
Data loss will occur if the module is deleted

See also

[System Requirements](#)

[Home](#) > [Getting Started](#)

## Getting Started

Topics in this section



[About Form Master](#)



[Form Submit Workflow](#)



[Form Elements](#)



[Menu Items](#)

See also

[Welcome](#)  
[Installation](#)  
[Settings](#)  
[Basic Functions](#)  
[Data Manager](#)  
[Formatting](#)  
[Reports](#)  
[Messaging](#)  
[Advanced Functions](#)

## About Form Master

Form Master 2008 is a form creation module for the DotNetNuke® Content Management System. Form Master 2008 allows an average user to create a wide variety of input forms comprising all the common Input Controls available for any web page quickly and easily. As needs become more complex, Form Master 2008 provides the advanced user with the flexibility and tools to include a high degree of functionality in one complete package. Listed below are the functional areas which will be discussed in more detail later in this User Guide.

1. Form Creation
2. Input Validation
3. Data Storage
4. Email
5. Auto Responder
6. Post Submission Options
7. Data Retrieval
8. User Registration

### See also

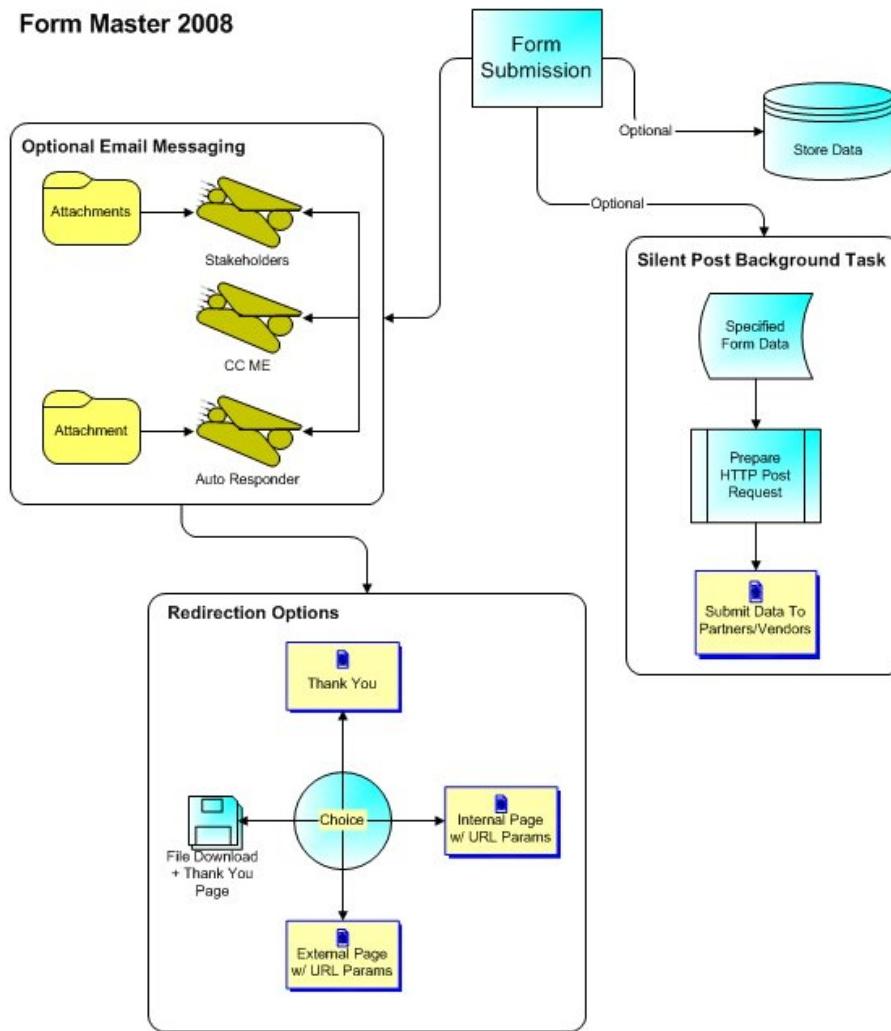
[Form Submit Workflow](#)  
[Form Elements](#)  
[Menu Items](#)

[Home](#) > [Getting Started](#) > [Form Submit Workflow](#)

## Form Submit Workflow



### Form Master 2008



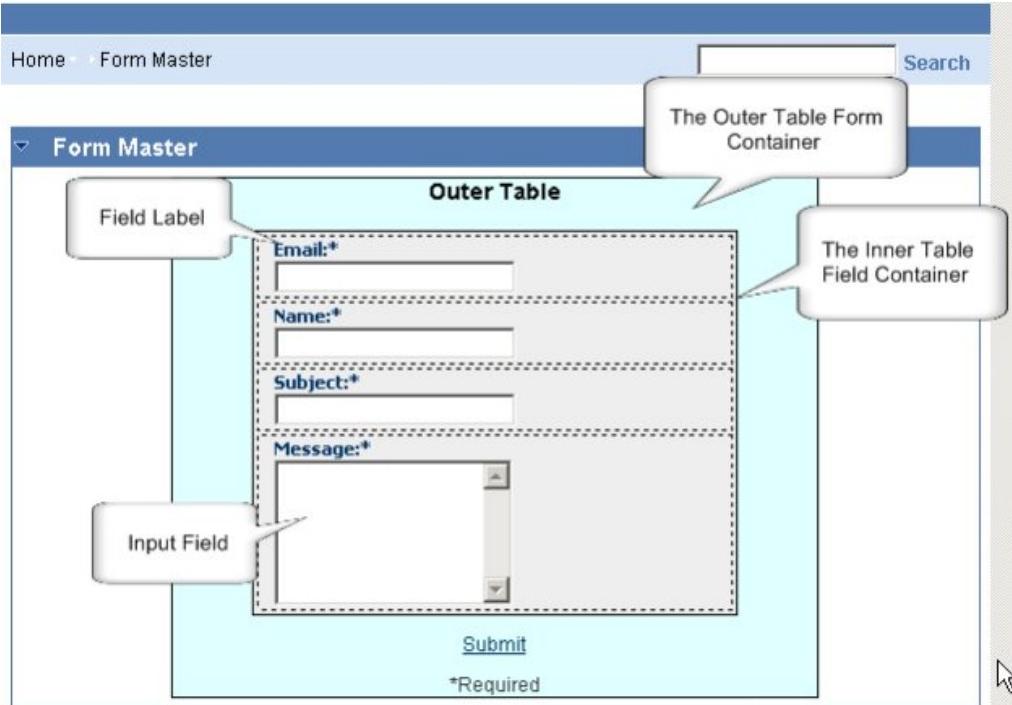
### See also

[About Form Master](#)  
[Form Elements](#)  
[Menu Items](#)

Home > Getting Started > Form Elements

## Form Elements

Form Master end user interface is made up of several HTML elements which will be referred to in other sections of the User Guide.



See also

[About Form Master](#)  
[Form Submit Workflow](#)  
[Menu Items](#)

[Home](#) > [Getting Started](#) > [Menu Items](#)

## Menu Items

- Form
  - Edit Form: This is where you will spend the majority of time. This section provides the tools to design the form
  - Form Query: The Form Query Editor allows for Getting Data from Database Tables, Spreadsheets, Text Files and other Data Sources supported by ADO.NET. This data can then be used to Auto Fill fields on the form based on the query Where clause. Note: Form Query is available for Super Users and users in the Administrators Role only.
  - Field Rules: The Field Rules page allows you to define rules which will show or hide fields on the form based on the value of other fields on the form. For example if Region is Other then show the Other Text Box field
- Settings:
  - Form Settings: Form Name, Email, Database, and Captcha settings
  - Other Settings: Other specific functionality settings can be seen
  - Web Service Settings: Used to setup Authorization and get the URL for the Web Service. Note: Web Service Settings is available for Super Users and users in the Administrators Role only.
  - Data Manager: The Data Manager allows for Mapping Existing Fields to a Data Table, Create Table from Form, or Create Form from Table. Note: Data Manager is available for Super Users and users in the Administrators Role only.
  - User Manager: The User Manager page allows you to Register a User and include them in multiple Roles when the Form is Submitted. Users who are already Logged in, can be added to Roles. Note: User Manager is available for Super Users and users in the Administrators Role only.
- Files: Used to View and Manage uploaded Files
- Messaging:
  - Form Heading: Used for creating the form heading or introduction above the start of the form entry fields
  - Form Thank You: Used for creating the message displayed after the user submits the form
  - Auto Responder: Used for designing the Auto Responder Email
  - Custom Template: Used for designing the Custom Email Template
- Report: Used for viewing or downloading collected data
- Exit: Leave the Form Master Edit menu
- User Guide: A link to the online User Guide

### See also

[About Form Master](#)  
[Form Submit Workflow](#)  
[Form Elements](#)

---

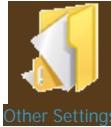
Home > Settings

## Settings

Topics in this section



Form Settings



Other Settings

Settings in Form Master are Controlled in the "Form Settings" page and the "Other Settings Page"

See also

Welcome  
Installation  
Getting Started  
Basic Functions  
Data Manager  
Formatting  
Reports  
Messaging  
Advanced Functions

## Form Settings

### Form Settings

The Form Settings page allows for setting options for email, saving of data etc. Shown below is the form with a description of each field below. The setting here are saved when the form is exported using the Export menu item in the module action menu.

<b>1</b>	<b>Form Name:</b> Contact Us
<b>2</b>	<b>Table Class:</b>
<b>3</b>	<b>Send To Action:</b> Use Admin Email
<b>4</b>	<b>To Email:</b>

1. Form Name: The form name is used in the Subject for emails, and for the Cookie name when the form is in Survey Mode. Form names should be unique for each domain name for the Survey Mode feature to work properly.
2. Table Class: The class name for the "Inner Table" in the Elements section above.
3. Send to Action: There are four options described below:
  - o None: No email is sent when the form is submitted
  - o Use Admin Email: Email is sent to the Site Administrator when form is submitted
  - o User Defined: Email is sent to the email address entered in the "To Email:" textbox. Multiple recipients can be entered separated by a semi-colon.
  - o Use Form Value: Email is sent to the email address in a selected form field. This is useful for example: A Dropdown List could contain Regions with the value of each region being an email address of the person responsible for covering that region.
4. To Email: Depending on the value of the "Send To Action" this can be:
  - o A Disabled Textbox
  - o An Enabled Textbox
  - o A Dropdown list of Fields defined on the form

<b>5</b>	<b>Send From Action:</b> Use Admin Email
<b>6</b>	<b>From Email:</b>
<b>7</b>	<b>Show CC Me on Form:</b> <input type="checkbox"/>
<b>8</b>	<b>CC Me Email:</b>
<b>9</b>	<b>BCC:</b>

5. Send From Action: This is important when the recipient of the email hits Reply in their Email client. There are three options described below:
  - o Use Admin Email: The email is send from the Site Administrator
  - o User Defined: Email is sent from the email address entered in the "From Email:" textbox
  - o Use Form Value: Email is sent from the email address in a selected form field.
6. From Email: Depending on the value of the "Send From Action" this can be the same as the "To Email" field in #5 above
7. Show CC Me on Form: This will enable/disable the checkbox at the bottom of the form so users can be Carbon Copied on the email which is sent.
8. CC Me Email: When "Show CC Me on Form" is checked allows for the selection of the form field to use to send a copy to the user.
9. BCC: Use the Blind Carbon Copy to send an email to any email address entered in the Textbox. Multiple recipients can be entered separated by a semi-colon.

**10**  Save Data to Database

**11**  Use Captcha

 Captcha Characters:

**12**

Update

10. Save Data to Database: When checked, saves form data to the database for later retrieval
11. Use Captcha: Check to Enable the use of the DotNetNuke Captcha Requirement on the Form
12. When "Use Captcha" is Checked, specify the Characters which will be used in the Captcha Image (optional)

#### See also

[Other Settings](#)

## Other Settings

### Basic Settings

Main Form Table Width:

Main Form Table Class:

Form Table Width:

Form Table Height:

Main Form Table Alignment:

Survey Mode

Allow Anonymous Uploads

Authenticate File Downloads

Hide Posted By in reports

Use Form Field Value for Email Subject

Second Field Value to Append to Subject:

Silent Post Url:

- Main Form Table Width: This is the width in Pixels or Percentage for the "Outer Table" shown in the Elements Section above
- Main Form Table Class: This it the class name for the "Outer Table" shown in the Elements Section above.
- Form Table Width: The width of the table which contains all the controls. This table is inside the Main Table which contains the Header text in the First Row and the Form Table in the Second Row. This value can be a whole number or a percentage.
- Form Table Height: The height of the table which contains all the controls. This table is inside the Main Table which contains the Header text in the First Row and the Form Table in the Second Row. This value can be a whole number or a percentage. Percentage does not work with all browsers.
- Outer Table Alignment: This is the Outer Table which contains the Heading Message in the top row and the Form in the second row.
- Inner Table Alignment: This is the Inner Table which directly contains the fields and is contained in the Outer Table
- Survey Mode: When checked prevents users from submitting the form more than once. When using this mode, the "Redirect after Submit" must be set to a value other than "None". Default un-checked.
- Allow Anonymous uploads: When unchecked any File Upload field will be disabled for anonymous users. Default un-checked.
- Authenticate File Downloads: When a file is uploaded, a link is created in the email to the downloaded file. When this is checked, only users with Edit privileges on the module will be allowed to download the file. Default is checked.
- Hide Posted By in Reports: By default Form Master will add the Submitted by column to the posted data and in the email. Checking this option removes the data. Default un-checked.
- Use Form Field for Email Subject: When checked, the Field Value specified in the Dropdown List will be used as the Subject line
- Second Field Valut for Email Subject: Optionally Select a Second Field Value to Apend to the Subject Line selected in #14
- Silent Post Url: Url to which a Silent Post of Form Data is made on Submit.

## Auto Responder Settings

 **Send Auto Responder Using**

 **Auto Responder From Email:**

 **Auto Response Subject:**

 **Auto Responder File Attachment**

**File Location:**

**File Name:**

[Upload New File](#)

- Send Auto Responder User: If Checked, the form will send an Email Auto Responder using the Field value selected in the Dropdown List.
- Auto Responder From Email: Specify the Auto Responder From Email address. For Example ([s.provencher@code5systems.com](mailto:s.provencher@code5systems.com)).
- Auto Response Subject: Specify the Email Subject for the Auto Responder Subject Line
- Auto Responder File Attachment: Specify a File to be attached to the Auto Responder Email. Usefull for Softare Trials etc.

## Redirect After Submit Options

 **Redirection After Submit**

**Link Type:**

- None**
- URL ( A Link To An External Resource )**
- Page ( A Page On Your Site )**
- File ( A File On Your Site )**

**Select A Web Page From Your Site:**

- Redirection After Submit: There are four options for redirection described below:
  - None: No redirection the Thank You page is shown by default
  - URL: This allows the user to enter any valid Internet Address.
  - Page: This allows the user to select a page within the site
  - File: This allows the user to select a file on the server for download to the user's browser. When this option is used the default Thank You message is shown and for most Browsers (User Agents), the file download should automatically start. A link to the file is provided for those browsers which block automatic downloads.

## Email Template and Format Settings

 **Email Template Type**

- XSL Template**
- Text Template**
- Custom Template**

 **Text Template:**

 **Send HTML formatted Email**

 **Send Multipart Email**

 **Use custom Email XSL Stylesheet**

 **Upload Custom Email Stylesheet:**

**Browse...**

- Email Encoding : Form most installations using the Default value will work just fine. If there are problems with the email containing strange characters, experimenting with the other encodings may help with the problem.
- Email Template: Form Master offers two Template styles for formatting the email message which are described below:

- Text Template: The text templates can be used for "Plain Text", or "Html" formatting. For each format the file contains text with replaceable tokens to insert the form data.  
The files can be found in the Templates folder in the Module root. Copying these files to a new folder under the Templates folder allows for creating multiple templates on a Host.

To Make Templates which are Specific to Individual Portals Create a Folder Under the Templates Fold corresponding the the Portal's PortalID for example Templates/0

Each format is made up of three files with the .TXT extension for "Plain Text" and .HTM for "Html":

- HEADER
- ITEM
- FOOTER

- XSL Templates: The XSL templates are available for those users familiar with XSL style sheet transformations. For version 1.1, this was the default method for formatting the email message.

The XSL Templates offer the advantage of being able to supply a unique custom template for each instance of Form Master without affecting any other instance. A sample data document located as "assets/xml/SampleData.xml" from the module root can be used to develop the XSL style sheet.

The "Text", or "Html" format default style sheets can be found at "assets/xslt", EmailSend.xslt and EmailSendHtml.xslt. It is recommended that not to overwrite these files, but rather upload a custom style sheet using the "Upload Custom Email Style Sheet" option.

- Custom Template: A Custom Template can be designed using the HTML Editor and the Field Tokens defined when creating the Form. This Template type is by Default sent in HTML format.
- Text Template: Only "default" is available in the Dropdown List until new templates are added by copying the default files to a new folder as noted in 19.1 above. When new templates are available the folder name will appear in the Dropdown List.
- Send HTML formatted Email: By default Form Master sends email in "Plain Text" format, If "Html" format is needed, simply check this box, and the default template with be used, for either style of template. (Custom Template is HTML always)
- Send Multipart Email: Check to Send HTML email with a Text Section for Email Readers which cannot choose not to show HTML (Note: Only applies to HTML emails) Checked by Default.
- Use custom Email XSL Stylesheet: To use a custom XSL style sheet check this checkbox.
- Upload custom Email Stylesheet: File Upload to upload a custom style sheet when #22 is checked

## Validation Options

<input checked="" type="checkbox"/>	 <b>Enable Client Script</b>
<input type="checkbox"/>	 <b>Use Validation Summary</b>
<b>Summary Heading Text:</b>	
<input type="text"/>	
<b>Summary Heading Text:</b>	
<input type="text"/> NormalRed	
<input checked="" type="checkbox"/>	 <b>Show Validation Summary</b>
<input type="checkbox"/>	 <b>Show Validation Summary Message Box</b>
<input type="checkbox"/>	 <b>Supress Field Validation Messages</b>
<b>Suppressed Field Character</b>	
<input type="text"/> *	
<b>Validation Summary Mode:</b>	
<input checked="" type="radio"/> <b>List</b> <input type="radio"/> <b>Bullet List</b> <input type="radio"/> <b>Single Paragraph</b>	
<b>Validation Summary Row:</b>	
<input type="button" value="Top"/>	
<b>Validation Summary Cell Alignment</b>	
<input checked="" type="radio"/> <b>Left</b> <input type="radio"/> <b>Center</b> <input type="radio"/> <b>Right</b>	

### Validation Summary Video

- Enable Client Script: When checked, enables field validation in the user's browser. When un-checked all validation is done on the Server which forces a reload of the page.
- Use Validation Summary: Check to use a Validation Summary Control on the Form. A Validation Summary shows all the Validation Errors in one central location and/or in an Alert popup box
- Summary Heading Text: The Text which will appear before any error messages, for example "Errors Have Occurred"
- Validation Summary CSS Class: The CSS class which is used for the displayed Error Messages
- Show Validation Summary: Check to show the Validation Summary on the Page when a Validation Error Occurs
- Show Validation Summary Message Box: Check to show an Alert Box when Form Validation Fails
- Supress Field Validation Messages: When using the Validation Summary, check this option to stop the Validation Messages from showing under the Fields
- Suppressed Field Character: When Supress Field Validation Messages is On. This value will be output to the right of the Field. This value can be any

character, string, or an Image Tag

Example: 

- Validation Summary Mode: Error Messages can be displayed in three modes: List, Bullet List, or Paragraph
- Validation Summary Row: Generally Validation Summary is placed at the Top or the Bottom of a form. Form Master offers the option to place the Summary at any position
- Validation Summary Cell Alignment: Select the Cell Alignment for the Error Messaging

See also

[Form Settings](#)

[Home](#) > [Basic Functions](#)

## Basic Functions

Topics in this section



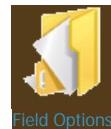
Form Layout



Tab Index



Adding Fields



Field Options



Validation



Localization



File Management

Add Content

See also

[Welcome](#)  
[Installation](#)  
[Getting Started](#)  
[Settings](#)  
[Data Manager](#)  
[Formatting](#)  
[Reports](#)  
[Messaging](#)  
[Advanced Functions](#)

Common Options Home > Basic Functions > Form Layout

## Form Layout

The Form Master 2008 Form Layout page allows you to add, delete, and Move fields in a visual designer. The designer includes several options which are described following the Screenshots shown below.



- **Label Position:** Specify where the Label for the Field will be shown. Options are:
  - Above: The text in the **Label Field** will be output above the Field in the same Table Cell
  - Left: The text in the Label Field will be output to the Left of the Field in a separate Table Cell
- **Cols:** The Number of columns in the form table
- **Rows:** The Number of rows in the form table
- **Update:** Click the Update link to save changes to the settings listed above.
- **Reset Form:** Click the "Reset Form" link to delete all Fields from the form. Note: This cannot be undone.

This screenshot shows a visual representation of a form layout with three rows:

- R:1:** Contains a field labeled "Email:\*" with a red validation message: "Please enter an email Email format is invalid". Below the field are "Edit", "Del", and "Move" links. Below the row are CS: 1, RS: 1, and TI: 1 dropdown menus.
- R:2:** Contains a single "Add" link.
- R:3:** Contains a field labeled "Name:\*" with a red validation message: "Please enter a name". Below the field are "Edit", "Del", and "Move" links. Below the row are CS: 1, RS: 1, and TI: 2 dropdown menus.

The top row of the form layout contains "C:" and the Column number. Clicking on for example "C:1" opens the Column Manager which will allow you to Insert or Delete Columns in the form.



The first Column of the form layout contains "R:" and the Row Number. Clicking on for example "R:1" opens the Row Manager which will allow you to Insert or Delete Rows in the form.

- **Add:** The "Add" link opens the Input Control Form where a new Form Field can be added.
- **Edit:** The "Edit" link allows for editing a Form Field.
- **Move:** The "Move" link allows for moving a Field to another location on the form.
- **CS:** Column Span for the Field.
- **RS:** Rowspan for the Field.
- **TI:** Tab Index for the Field. When the user uses the Tab key to move between fields on a form the TI dropdown list can be used to control the order of the fields the will be move to when the Tab Key if pressed

[View the Form Layout Video](#)

See also

[Tab Index](#)  
[Adding Fields](#)  
[Field Options](#)  
[Validation](#)  
[Localization](#)  
[File Management](#)

---

[Home](#) > [Basic Functions](#) > [Tab Index](#)

## Tab Index

The Tab Index Feature allows for the ordering of the input focus when the user presses the "Tab" key on their keyboard

The Tab Index is set by using the "TI:" dropdown in each field cell while in design mode. Label fields do not participate in the Tab Index.

The Ordering can be Left-Right, then Top-Down.

[Setting Left-Right then Top-Down Tab Index Presentation](#)

Or Top-Down, then Left-Right

[Setting Top-Down, then Left-Right Tab Index Presentation](#)

In reality, the ordering can be in any combination, these shown are the most common.

### See also

[Form Layout](#)  
[Adding Fields](#)  
[Field Options](#)  
[Validation](#)  
[Localization](#)  
[File Management](#)

[Home](#) > [Basic Functions](#) > [Adding Fields](#)

## Adding Fields

Topics in this section



Common Options



Textbox



Checkbox



Dropdown List



Listbox



Checkbox List



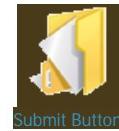
Radiobutton List



Label



File Upload



Submit Button



Hidden Field

See also

[Form Layout](#)  
[Tab Index](#)  
[Field Options](#)  
[Validation](#)  
[Localization](#)  
[File Management](#)

## Common Options

Common Options are common to all types of Form Fields i.e the Textbox, Dropdown List, etc.

**Input Control**

<b>Type:</b>	<input style="width: 100px; height: 20px; border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;" type="button" value="TextBox"/>		
<b>Label:</b>	<input style="width: 450px; height: 100px; border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;" type="text"/>		
<b>Field Name:</b>	<input style="width: 150px; height: 20px; border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;" type="text"/>	<b>Width:</b>	<input style="width: 150px; height: 20px; border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;" type="text"/>
<b>Label Align:</b>	<input style="width: 100px; height: 20px; border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;" type="button" value="NotSet"/>	<b>Control Align:</b>	<input style="width: 100px; height: 20px; border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;" type="button" value="NotSet"/>
<b>Label Class:</b>	<input style="width: 150px; height: 20px; border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;" type="text" value="SubHead"/>	<b>Height:</b>	<input style="width: 150px; height: 20px; border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;" type="text"/>
<b>Disabled:</b>	<input type="checkbox"/> Disable User Input	<b>Hide Label:</b>	<input type="checkbox"/> Hide Label
<b>Field Token:</b>	<input style="width: 150px; height: 20px; border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;" type="text"/>	<b>Exclude Field:</b>	<input type="checkbox"/> Exclude from Email & Data
<b>Allow Markup:</b>	<input type="checkbox"/> Allow Markup (html)	<b>Class:</b>	<input style="width: 150px; height: 20px; border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;" type="text" value="NormalTextBox"/>

- Type: Select the type of field needed
  - Textbox
  - Checkbox
  - DropDownList
  - ListBox
  - CheckBoxList
  - RadioButtonList
  - Label
  - File Upload
  - Submit Button
  - Hidden Field
- Label: The visible Label for the input field.
- Field Name: The Name which will be used in the Database Columns, and for Columns in Reports
- Label Align: The alignment of the Label in the Table Cell when Label Position is on Left
  - Not Set
  - Left
  - Center
  - Right
- Label Class: The CSS class applied to the Label Default is DotNetNuke "SubHead" class. Portal.css is the best place to define CSS Class Rules
- Disabled: Disable User input ,useful if user need to see pre-filled data but is not allowed to change it
- Field Token: A Replaceable Token which is replaced by the Submitted Data in Email messaging and the Thank You message. Usually formatted like [TOKENNAME]
- Encrypt Field: Useful for hiding Data on the Web Page such as a List of Email Addresses which are used for the Send To: address on a Lead Generation Form
- Allow Markup: Check to allow markup (html tags) in field input.
- Width: Specify the Width of the input control in Pixels. Example: 250
- Controls Align: The alignment of the input control in the Table Cell
  - Not Set
  - Left
  - Center
  - Right
- Height: Specify the Height of the input control in Pixels. Example: 150
- Hide Label: Hides the Label specified in #1. This can be useful when a actual Label Control is used as a heading for a CheckBoxList for example.
- Exclude Field: Exclude the field from being sent in the Email & Stored in the Database. This is useful when the Silent Post Feature is being used and it is not necessary to store or email required Hidden fields used in the Silent Post
- Class: The CSS class applied to the input control. Default is DotNetNuke "NormalTextBox". This field allows for a second Class name separated by a semi-colon ";" which when present is applied to the Table Cell containing the input control. Portal.css is the best place to define CSS Class Rules

See also

Textbox  
Checkbox  
Dropdown List  
Listbox  
Checkbox List  
Radiobutton List  
Label  
File Upload  
Submit Button  
Hidden Field

## Textbox

The Textbox is the workhorse of most any form, and adding a Textbox only takes a few mouse clicks and keystrokes. In addition to adding a Textbox, this presentation will introduce:

- Adding rows to the form
- Entering a Label for a field
- Using the Auto-Fill feature
- Introduce the field management controls

[Add Textbox Presentation](#)

### See also

[Common Options](#)  
[Checkbox](#)  
[Dropdown List](#)  
[Listbox](#)  
[Checkbox List](#)  
[Radiobutton List](#)  
[Label](#)  
[File Upload](#)  
[Submit Button](#)  
[Hidden Field](#)

[Home](#) > [Basic Functions](#) > [Adding Fields](#) > [Checkbox](#)

## Checkbox

The Checkbox is a very simple field control to add to the form. In addition to adding a Checkbox, this presentation will introduce:

- Setting a default for a field control
- Hiding the field label
- Positioning the field labels to the left of the fields on the form

For the Checkbox the “Label Class” is used to format the text.

[Add Checkbox Presentation](#)

### See also

[Common Options](#)  
[Textbox](#)  
[Dropdown List](#)  
[Listbox](#)  
[Checkbox List](#)  
[Radiobutton List](#)  
[Label](#)  
[File Upload](#)  
[Submit Button](#)  
[Hidden Field](#)

## Dropdown List

The Dropdown List and all the List type field controls share one common element which is the "List Item".

A List Item is what appears in the Dropdown List when you click on the down arrow. A List Item is made up of two parts the "Text" and the "Value" parts. These two values can be the same, or as an example in a list of US states "New York" will be the Text part and "NY" will be the Value part.

With the Dropdown List, it is always a best practice to enter a Prompt for the user as the first item in the list for example: Enter "<< Select >>" for the Text part and "0" for the Value part. This will ensure that Validation can be applied to the field forcing the user to make a selection from the list. Without Validation, users will often submit the form without making a selection, submitting the first item in the Dropdown List as default.

Adding the Dropdown List will introduce new features, and make use of some which have already been covered:

- Use of the Auto-Fill feature which in the case of the Dropdown will find the correct value in the list and pre-select that for the user
- Manually adding a Item to the List Items form
- Adding List Items by using the Fill List Items with States from a selected Country feature

We will add a Required Field Validator to this Dropdown List in the Validation section.

[Add DropDown List Presentation](#)

### See also

[Common Options](#)  
[Textbox](#)  
[Checkbox](#)  
[Listbox](#)  
[Checkbox List](#)  
[Radiobutton List](#)  
[Label](#)  
[File Upload](#)  
[Submit Button](#)  
[Hidden Field](#)

## Listbox

The Listbox control is useful where there are a small number of items in a list for the user to choose from.

Typically, a Listbox can be in two modes, Single, or Multi select modes. By default, Form Master uses Multi select mode only. The reason for this is simple, for Single selection mode, a Dropdown List, or RadioButton List provide a better user interface.

For Multi select mode, the preferred field control is the Checkbox List which offers a better user interface as there is no need for the user to know how to use "Ctrl Click" Selects Multiple Items functionality.

The one advantage in using a Listbox is when a user must make a selection, as the Checkbox List cannot have a Required Field Validator, whereas the Listbox can.

In addition to adding a Listbox, this presentation will introduce:

- Adding a Column to the form
- Moving Controls between cells

[Add Listbox Presentation](#)

### See also

Common Options  
Textbox  
Checkbox  
Dropdown List  
Checkbox List  
RadioButton List  
Label  
File Upload  
Submit Button  
Hidden Field

## Checkbox List

This opening paragraph should describe the feature that you are documenting. Explain how it is commonly used and what the benefits are. For example: The Widget Master email link allows you to easily send information about each widget to various departments within your company. Often, the feature that you are documenting can be best explained by walking the reader through step by step. Use screenshots to illustrate the steps where possible.

1. Start the application by...
2. On the startup screen, click the...
- 3.

### See also

[Common Options](#)  
[Textbox](#)  
[Checkbox](#)  
[Dropdown List](#)  
[Listbox](#)  
[Radiobutton List](#)  
[Label](#)  
[File Upload](#)  
[Submit Button](#)  
[Hidden Field](#)

## Radiobutton List

The RadioButton List is used when the number of items in the list is small, and the user is allowed to select one item in the list. The procedure for adding a RadioButton List is the same as for the Checkbox List. The only two differences are:

1. There can only be one selected item in the List Items form, whereas the Checkbox List can have multiple items selected in the List Items form.♦
2. The Required Field Validator can be used to require that a user make a selection.

We will add a Required Field Validator to this RadioButton List in the Validation section

[Add RadioButton List Presentation](#)

### See also

[Common Options](#)  
[Textbox](#)  
[Checkbox](#)  
[Dropdown List](#)  
[Listbox](#)  
[Checkbox List](#)  
[Label](#)  
[File Upload](#)  
[Submit Button](#)  
[Hidden Field](#)

## Label

The Label Control can be used to insert any Html content including pictures to a Table Cell in the Form. The most likely use will be to introduce a section of the form or to provide instructions to the user which involve more than a one to three word prompt entered in the Label textbox on the "Input Control" form.

When used to provide instructions it is common to hide the label prompt for the target field control so as not to duplicate the field name.

We will use the Label Control to introduce and provide instructions for the "Interests", and "Favorite Color" fields added above. This will require us to move these two fields in the form to make room for the Label above.

[Add Label Control Presentation](#)

### See also

[Common Options](#)  
[Textbox](#)  
[Checkbox](#)  
[Dropdown List](#)  
[Listbox](#)  
[Checkbox List](#)  
[Radiobutton List](#)  
[File Upload](#)  
[Submit Button](#)  
[Hidden Field](#)

[Home](#) > [Basic Functions](#) > [Adding Fields](#) > [File Upload](#)

## File Upload

The File Upload Field allows end users to upload files to a Server.

### File Storage

Files can be stored in the Database or in the File System. Even if the Form is not set to save the form submissions to the database the uploaded files are still stored in the database if the individual File Upload Field is set to that storage media. Multiple File Upload Fields can store the file in different locations, depending on the setting of the individual field.

### File Size\*

File size can be controlled by each field on the form, the default is 1000KB. In a form with multiple File Upload Fields, the module uses Transactional Processing to Rollback the entire form submission if any part of the submission fails, which would happen if a file were larger than the size set in the File Upload Field settings.

\*The default maximum size for all content uploaded to a server is set to 4096 KB by ASP.NET See Appendix A for instructions on modifying this value.

### File Types

Any File Type which the DNN Host has allowed can be uploaded using the File Upload Field. If a file type is not listed, it will be necessary to contact the DNN Host to add that extension to the allowable extensions for the DNN installation.

### Downloads

If emailing is enabled for the form a link to the uploaded file will be in the body of the email, also the file will be available for download from the Reports area of the Form Master Edit menu. By default Form Master requires that a user be in an Authorized Edit Role in order to download a file from the server, but Form Master can be set to allow anonymous file downloads if needed.

### Validation:

Form Master automatically adds a Regular Expression Validator for the File Upload Field to ensure that the File Type is one of the selected types. A Required Field Validator can be added when the end user must supply a file in order to submit the form.

[Add File Upload Presentation](#)

### See also

[Common Options](#)  
[Textbox](#)  
[Checkbox](#)  
[Dropdown List](#)  
[Listbox](#)  
[Checkbox List](#)  
[Radiobutton List](#)  
[Label](#)  
[Submit Button](#)  
[Hidden Field](#)

## Submit Button

The Submit Button comes in three types:

- Standard HTML Button
- Link Button
- Image Button

Form Master automatically adds a Link Button to the form if no Submit Button is added by the form designer.

Form Master supports Multiple Submit Buttons on a single form, and each Button can be a different type. Multiple Submit Buttons are useful for example on a large form where some fields are required and others are optional. Required fields can be grouped at the top of the form with a Submit Button below the group. Another Submit Button can be added at the bottom of the form after the optional fields.

[Adding Submit Buttons Presentation](#)

### See also

[Common Options](#)  
[Textbox](#)  
[Checkbox](#)  
[Dropdown List](#)  
[Listbox](#)  
[Checkbox List](#)  
[Radiobutton List](#)  
[Label](#)  
[File Upload](#)  
[Hidden Field](#)

## Hidden Field

The Hidden Field has many uses on a Web form. It can be used to provide data to the Database, Email Recipient, or Process which the user does not or should not see.

Consider a scenario where the user will be redirected to an affiliate web site after the Form Submission. A requirement of course in these cases is an Affiliate ID in the URL Querystring. Using the Hidden Field, an Affiliate ID can be added to the form which the user will not see. When setting up this field, make sure that Querystring Parameter name is what is expected by the Affiliate and mark this field to be sent by Url in the Send Value By Options of the Input Control Page. In this scenario, it is also necessary to make sure the form redirects after submission in the Settings > **Other Settings** page. It is also common to exclude Hidden Fields used in this way from being stored or sent in the email using the Exclude from Email & Data option as shown below.

Hidden Field Screen Capture

**Input Control**

<b>Type:</b>	<input type="button" value="Hidden Field"/>		
<b>Label:</b>	AffiliateID		
<b>Field Name:</b>	AffiliateID	<b>Width:</b>	
<b>Label Align:</b>	NotSet	<b>Control Align:</b>	NotSet
<b>Label Class:</b>	SubHead	<b>Height:</b>	
<b>Disabled:</b>	<input type="checkbox"/> Disable User Input	<b>Hide Label:</b>	<input type="checkbox"/> Hide Label
<b>Field Token:</b>		<b>Exclude Field:</b>	<input checked="" type="checkbox"/> Exclude from Email & Data
<b>Encrypt Field</b>	<input type="checkbox"/> Encrypt Field	<b>Class:</b>	NormalTextBox

**Auto Fill Options**

<b>Auto Fill:</b>	<input checked="" type="radio"/> None	<input type="radio"/> Membership	<input type="radio"/> Session	<input type="radio"/> Url	<input type="radio"/> Cookie	<input type="radio"/> Form Query	<input type="radio"/> Field Query
<b>Default Text:</b>	123456 <-- This is the Affiliate ID						

**Field Query Options**

**Send Value By Options**

<b>Send Value By:</b>	<input type="radio"/> None	<input type="radio"/> Session	<input checked="" type="radio"/> Url	<b>Value Name:</b>	<input type="text" value="affid"/>
	<input type="radio"/> Cookie	<input type="radio"/> Silent Post	<input type="radio"/> Silent Post & Session		
	<input type="radio"/> Silent Post & Cookie				

See also

[Common Options](#)  
[Textbox](#)  
[Checkbox](#)  
[Dropdown List](#)  
[Listbox](#)  
[Checkbox List](#)

Radiobutton List  
Label  
File Upload  
Submit Button

[Home](#) > [Basic Functions](#) > [Field Options](#)

## Field Options

Topics in this section



[Auto Fill Options](#)



[Textbox Options](#)



[Field Query Options](#)



[Send Value By Options](#)

See also

[Form Layout](#)

[Tab Index](#)

[Adding Fields](#)

[Validation](#)

[Localization](#)

[File Management](#)

Home > Basic Functions > Field Options > Auto Fill Options

## Auto Fill Options

<b>A</b>	
<b>1</b>	<input checked="" type="radio"/> Auto Fill: <input checked="" type="radio"/> None <input type="radio"/> Membership <input type="radio"/> Session <input type="radio"/> Url <input type="radio"/> Cookie <input type="radio"/> Form Query <input type="radio"/> Field Query
<b>2</b>	<input checked="" type="checkbox"/> Encrypt Field <input type="checkbox"/> Encrypt Field
<b>3</b>	<input checked="" type="checkbox"/> Default Text: <input type="text"/>
<b>B</b>	
<b>4</b>	<input checked="" type="radio"/> Auto Fill: <input type="radio"/> None <input checked="" type="radio"/> Membership <input type="radio"/> Session <input type="radio"/> Url <input type="radio"/> Cookie <input type="radio"/> Form Query <input type="radio"/> Field Query
<b>5</b>	<input checked="" type="checkbox"/> Encrypt Field <input type="checkbox"/> Encrypt Field
<b>6</b>	<input checked="" type="checkbox"/> Auto Fill From: <input type="text"/> << Select >> <input type="button" value="▼"/>
<b>C</b>	
<b>7</b>	<input checked="" type="radio"/> Auto Fill: <input type="radio"/> None <input type="radio"/> Membership <input checked="" type="radio"/> Session <input type="radio"/> Url <input type="radio"/> Cookie <input type="radio"/> Form Query <input type="radio"/> Field Query
<b>8</b>	<input checked="" type="checkbox"/> Encrypt Field <input type="checkbox"/> Encrypt Field
<b>9</b>	<input checked="" type="checkbox"/> Auto Fill Value: <input type="text"/>

1. Select the Auto Fill Option A - G
2. Encrypt Field: Useful for hiding Data on the Web Page such as a List of Email Addresses which are used for the Send To: address on a Lead Generation Form Note: This does not Encrypt the data when posted to the server, it only protects sensitive data which is output to the page before the user submits the form.
3. When 1.A "None" is selected, text entered here will be used in the appropriate way for the selected field type.
4. When 1.B "Membership" is selected, the dropdown list is shown and populated with Membership field names. This list will include custom Membership fields as well as the default DotNetNuke fields
5. When 1.C thru 1.G are selected this field contains:
  - o C: Session Variable Name
  - o D: URL Parameter Name
  - o E: Cookie Name
  - o F and G: Column Name

### See also

[Textbox Options](#)  
[Field Query Options](#)  
[Send Value By Options](#)

## Textbox Options

### **Text Box Options**

---

<input checked="" type="radio"/> <b>Text Mode:</b>	<input type="button" value="SingleLine"/> <input checked="" type="radio"/>	<input checked="" type="radio"/> <b>Wrap Text:</b>	<input type="checkbox"/> <b>Wrap in Text Box</b>
<input checked="" type="radio"/> <b>Columns:</b>	<input type="text"/>	<input checked="" type="radio"/> <b>Rows:</b>	<input type="text"/>
<input checked="" type="radio"/> <b>Use Calendar:</b>	<input type="checkbox"/> <b>Use Calendar</b>	<input checked="" type="radio"/> <b>Max Length:</b>	<input type="text"/>

1. Text Mode:
  1. Single Line: this mode is for standard one line data entry such as a name or street address
  2. MultiLine: this mode is for free form text such as a detaild problem report, or free form question
  3. Password: this mode allows for the user to securly enter a password
2. Columns: is an older style method for setting a Text Box width, the prefered method is by using the Width field
3. Use Calendar: This will add the default DotNetNuke Calendar functionality to the Form Field
4. Wrap Text: Check this to automatically Wrap text as the User types in a Multiline Textbox
5. Rows: Rows is an older style method for setting a Text Box height when Text Mode is MultiLine, the prefered method is by using the Height field
6. Max Length: Maximum number of characters allowed in the input field. This is not effecive for MultiLine Text Mode

### See also

[Auto Fill Options](#)  
[Field Query Options](#)  
[Send Value By Options](#)

## Field Query Options

Note: Field Query Options is available for Super Users and users in the Administrators Role only.

**About The Field Query**

**Connection String:** [Blank Input Field]

**SQL Query:**  
SELECT Value, Text FROM Lists WHERE ListName = 'Frequency'

[Test Query](#)

**Query Tokens**

#URL:YOURURLPARAMNAME#

Value	Text
D	Day(s)
M	Month(s)
N	None
O	One-time Fee
W	Week(s)
Y	Year(s)

**Text Column:** [Blank Input Field]

**Value Column:** [Blank Input Field]

- Connection String: The connection string is optional, and when left blank the Query Engine will use the default DotNetNuke Connection string defined in the Web.Config. The Query Engine will accept any Valid Connecton string which is Suported by ADO.NET. [ConnectionsStrings.com](#) is a great resource for connection string syntax. Connection Strings are Encrypted when stored for later use hiding any login information which might be used. It is advised that when defining a connection string with login information, the login be allowed only the premissions required to execute the Query.
- SQL Query: This is where the Query is Entered. Execution of Stored Procedures is supported. Writing Queries and Structured Query Language usage is beyond the scope if this user guide.
- Query Tokens: Replaceable Tokens from Membership can be used in Queries. Place the cursor where the Token should be inserted in the Query, Select the Token, and click on the "Insert Token" button. Tokens must be used Unchanged to work. Session and URL variables can also be used in the Query and can be inserted from the Query Tokens Dropdown List the Syntax is: #URL:YOURURLPARAMETERNAME# and #SESSION:YOURSESSIONPARAMETERNAME#
- The Query Engine will display the results of the Query or an Error Message when "Test Query" is clicked as shown in the Grid of Frequencies
- Text Column: When using a List Item type Control, the Query Column name is entered here which will be the data shown to the user
- Value Column: When using a List Item type Control, the Query Column name is entered here which will be the data used for the Value part of the List Control
- Test Query: Click this Link to test the Query entered in the "SQL Query" field.

### See also

[Auto Fill Options](#)  
[Textbox Options](#)  
[Send Value By Options](#)

Home > Basic Functions > Field Options > Send Value By Options

## Send Value By Options

### Send Value By Options

---

**Send Value By:**  None  Session  Url  
 Cookie  Silent Post  Silent Post & Session  
 Silent Post & Cookie

**Value Name:**

- Send Value By: Select how the Value for this field should be handled after the user has submitted the form. Choices are explained below:
  - None: Do not send the value
  - Session: Creates a Session Variable using the name in the "Value Name" field. This makes the value available by Name to any process in the Website which can assess the Session Object. A good example of use is to send the user to another Form Master form where some of the First Form Data is placed in Session and used in the Second Form.
  - Url: This option will send the value of the field as a Parameter in a Url using the name in the "Value Name" field when redirect is set in [Other settings](#). One example of use it sending Product ID information to an external Shopping Cart, or Payment Processor such as PayPal®.
  - Cookie: Creates a Transient Cookie using the name in the "Value Name" field. This makes the value available by Name to any process in the Website which can read a Cookie Value.
  - Silent Post: Creates an HTTP Post as a Background process to the URL specified in [Other Settings](#). [Read more about using Silent Post](#)
  - Silent Post & Session: A combination of the two actions adding more flexibility for the post submit process
  - Silent Post & Cookie: A combination of the two actions adding more flexibility for the post submit process
- Value Name: Assigns a name to the Field Value by which other processes can access the value.

### See also

[Auto Fill Options](#)  
[Textbox Options](#)  
[Field Query Options](#)

## Validation

### Topics in this section



Validation is used to ensure that users supply all the information required on the form, and that the data entered meets specific formatting requirements. A field can have only one of each type of Validator; generally two will be enough to do proper data validation.

During Design time, the error messages will be shown under the fields to show what validation rules have been applied. During Runtime, the error messages will be hidden until required to enforce validation.

Form Master makes use of three of the Validators available in ASP.NET

- Required Field Validator
- Regular Expression Validator
- Compare Validator

Additionally Form Master also used a Checklist Validator Control and the Validator Summary Control.

Field Validation is important to ensure that clean consistent data is collected which can be used efficiently throughout the organization

### See also

[Form Layout](#)  
[Tab Index](#)  
[Adding Fields](#)  
[Field Options](#)  
[Localization](#)  
[File Management](#)

## Required Field Validator

The Required Field Validator is used to ensure that a user enters a value in a Textbox, or selects a value from one of the "List Item" fields.

The Field Control types which support the Required Field Validator are:

- Textbox
- Dropdown List
- Listbox
- RadioButton List
- File Upload

The Required Field Validator has three attributes which can be modified when adding this Validator to a field:

- Error CSS Class: This is the CSS class name applied to the Error Message when displayed in the Browser. The DotNetNuke® class name "NormalRed", is used by default.
- Error Message: This is the message to be displayed in the Browser when the user input fails validation.
- Initial Value: The value entered here is used by the Required Field Validator to determine if the user supplied value is the same as the "Initial Value", if so then validation fails.

Using the Required Field Validator with the different field types is explained below:

### Adding a Required Field Validator to a Textbox or File Upload

Adding a Required Field Validator for a Textbox is straightforward, simply edit the field, click on "Add Validator" in the Validators Data Grid, modify the Error Messages, and click "Update". The Error Message is shown in the Designer.

### Adding a Required Field Validator to a Dropdown List

Adding a Required Field Validator to a Dropdown List requires some planning, as mentioned in Adding a Dropdown List, adding a prompt as the first item in the List Items data grid like "<< Select >>" for the Text part and "0" for the Value part gives us "0" to use as the Initial Value. If a user does not make a selection, validation will fail.

### Adding a Required Field Validator to a Listbox or RadioButton List

When an Initial Value is not used, adding a Required Field Validator to a Listbox or RadioButton list is the same as for the Textbox. When an Initial Value is used the steps are the same as for the Dropdown List.

This section will show how to require a user to agree to the Terms & Conditions in order to submit the form using a RadioButton List, but the steps would be the same for a Listbox. This section will also use a different form.

[Required Field Validator Presentation](#)

### See also

[Regular Expression Validator](#)  
[Compare Validator](#)  
[Checkbox List Validator](#)  
[Validation Summary](#)

## Regular Expression Validator

The Regular Expression Validator checks whether the value of an input control matches a pattern defined by a regular expression. This type of validation allows you to check for predictable sequences of characters, such as those in social security numbers, e-mail addresses, telephone numbers, and postal codes.

The Regular Expression Validator does not require that a field contain a value, it only validates if a value is present.

The use of Regular Expressions is beyond the scope of this User Guide, however there are many resources on the Internet to help learn how to create Regular Expressions for use in this Validator. Form Master provides 4 built-in expressions which are:

- E-Mail Address
- Internet URL
- U. S. Phone Number
- U. S. SSN
- U. S. or Canadian Postal Code

[Regular Expression Validator Presentation](#)

### See also

[Required Field Validator](#)  
[Compare Validator](#)  
[Checkbox List Validator](#)  
[Validation Summary](#)

[Home](#) > [Basic Functions](#) > [Validation](#) > [Compare Validator](#)

## Compare Validator

The Compare Validator is used to compare values in two different fields on the form, or compare the data entered to a specific type a data such as a Date.

The Compare Validator does not require that a field contain a value, it only validates if a value is present.

For example: Many forms ask the user to enter their email address then confirm the email address by entering it a second time to help the user avoid typographical errors. The Compare Validator is just the tool to use to solve this. The Compare Validator has four new attributes in the Validators Data Grid

- Operator: The values available here are:
  - **DataTypeCheck:** Use this when comparing the user entered value with a specific Data Type such as a Date
  - **Equal:** =; Compare value with another form field or with the Value to Compare. The remaining Operators share the same function.
  - **GreaterThan:** >
  - **GreaterThanOrEqualTo:** >=
  - **LessThan:** <
  - **LessThanOrEqualTo:** <=
  - **NotEqual:** <> or !=
- Value to Compare: When "Control to Compare" is "<< None >>", works with the Operator and Data Type to compare the user entered value.
- Control to Compare: Sets which form Field to Compare this Field to. Works with the Operator
- Data Type: When using the "DataTypeCheck" Operator forces the user to enter the proper type of data such as a Date or Whole Number

When the "Control to Compare" is not set to "<< None >>", the field will not be part of the data sent by email or saved to the database.

In this example, we will use the Compare Validator to make sure the user enters their email address without typographical errors.

[Compare to Field Validation Presentation](#)

This example validates a date entry field.

[Compare to Data Type Validation Presentation](#)

This example validates the value is Greater than or Equal to 5 and is a whole number

[Value to Compare Validation Presentation](#)

### See also

[Required Field Validator](#)  
[Regular Expression Validator](#)  
[Checkbox List Validator](#)  
[Validation Summary](#)

## Checkbox List Validator

The Checkbox List Validator can be used to make sure that a user checks a certain number of checkboxes. For Example we have a list of ten Options and a user must choose three of those Options to continue. See the Setup Screen Shot below.

Validators	
Validator Details	
<b>Validator Type:</b> ChkListValidator	<b>Error Message:</b> <input type="text" value="Please Select Three Options"/>
<b>Error Css Class:</b> <input type="text" value="NormalRed"/>	<b>Minumum Checked:</b> <input type="text" value="3"/>
<b>Maximum Checked:</b> <input type="text" value="3"/>	

[Update](#) [Cancel](#) [Delete](#)

### See also

[Required Field Validator](#)  
[Regular Expression Validator](#)  
[Compare Validator](#)  
[Validation Summary](#)

[Home](#) > [Basic Functions](#) > [Validation](#) > [Validation Summary](#)

## Validation Summary

The Validation Summary allows for locating all the Validation Error Messages in one place on the form or in an Alert Message Box or Both.

The Validation Summary is useful on Long forms where a Validation Error may have occurred on a field at the top of the form which has scrolled out of the user's view. Then this happens the user may think that the page is not working properly and leave.

Validation Summary Options Include:

- Placement Top, Bottom, or any row in between.
- Optional Header Text
- Simple List, Bullet List, or Paragraph style
- Message Box Alert
- Suppress the default Error Message under the Field
- Customizable Invalid Field Indicator when Suppress Error Message is set
- Left, Center, and Right cell alignment

[View the Video](#)

See also

[Required Field Validator](#)  
[Regular Expression Validator](#)  
[Compare Validator](#)  
[Checkbox List Validator](#)

## Localization

### Localization

Form Master supports localization using the standard DotNetNuke® methods. The App\_LocalResources folder in the Module root can be used to create Resource files for any Locale.

The files Resource files are listed below:

- FormAutoRespMsg.ascx.resx
- FormCustomTemplate.ascx.resx
- FormDataServiceSettings.ascx.resx
- FormDispatcher.ascx.resx
- FormFieldRules.ascx.resx
- FormHeadingEdit.ascx.resx
- FormMaster.ascx.resx
- FormMasterDBMgr.ascx.resx
- FormMasterEdit.ascx.resx
- FormMasterFiles.ascx.resx
- FormMasterFormOptions.ascx.resx
- FormMasterFormQuery.ascx.resx
- FormMasterOptions.ascx.resx
- FormReport.ascx.resx
- FormThankYouEdit.ascx.resx
- FormUserManager.ascx.resx
- SharedResources.resx

In addition the DotNetNuke Membership field names for the Auto-Fill feature can be Localized using the Lists menu option on the Host Menu.

Form Master does not support Multi-Lingual content at this time.

### See also

[Form Layout](#)  
[Tab Index](#)  
[Adding Fields](#)  
[Field Options](#)  
[Validation](#)  
[File Management](#)

[Home](#) > [Basic Functions](#) > [File Management](#)

## File Management

Form Master 2008 includes a File Manager page to manage the files uploaded by users.

The Files page allows for Downloading, Searching, and Deleting of uploaded files. Sorting is also available on the File ID, File, and Date columns.

When a file is uploaded when the "Save Data to Database", on the "Form Settings" is checked, the file cannot be deleted using the Files page since the files are linked to the data submitted by the user. These files are available here for download, but must be deleted with the related record

**Form Master Edit**

**Form    Settings    Files    Form Heading    Form Thank You    Report    Exit    User Guide**

**File Name:**  [Search](#) [Clear](#)

< Previous Page Next Page >				
Form Data ID	FileID	File	Date	Delete
0	33	DeleteMe.doc	2/28/2006 9:25:00 AM	<a href="#">Delete</a>
0	30	EN_DKU5.pdf	2/28/2006 9:22:00 AM	<a href="#">Delete</a>
0	24	HomePageIntro.doc	2/28/2006 8:41:00 AM	<a href="#">Delete</a>
0	35	Newsletter Project Scope.pdf	2/28/2006 9:27:00 AM	<a href="#">Delete</a>
1311	31	Q4NEReport.pdf	2/28/2006 9:22:00 AM	<a href="#">Delete</a>
0	32	SalesReps.ZIP	2/28/2006 9:24:00 AM	<a href="#">Delete</a>

< Previous Page Next Page >

**Files associated with a Data Record, must be deleted with that Record in Reports.**

### See also

[Form Layout](#)  
[Tab Index](#)  
[Adding Fields](#)  
[Field Options](#)  
[Validation](#)  
[Localization](#)

[Home](#) > [Data Manager](#)

## Data Manager

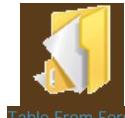
### Topics in this section



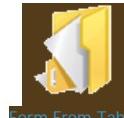
[The Connection String](#)



[Map Existing Table](#)



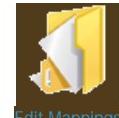
[Table From Form](#)



[Form From Table](#)



[The Mapped Form](#)



[Edit Mappings](#)



[External Table Mask](#)

Form Master 2008 Stores data as XML Fragments by default in a single Table Column. While convenient and easy to manage, this make it difficult for other applications to access the collected data. Form Master 2008 introduces the ability to store data in Standard Microsoft® SQL Server® Tables. Form Master 2008 offers three methods to help Map Form Fields to Standard SQL Tables as noted below.

Note: Data Manager is available for Super Users and users in the Administrators Role only.

- [Map an Existing Form to an Existing Table](#)
- [Create a SQL Table From a Form](#)
- [Create a Form From an SQL Table](#)

When Forms are Mapped to an SQL Table, Form Master 2008 will continue to store the data in XML Fragments as a backup. This gives the user the ability to restore the data to a SQL Table by re-mapping the form.

### See also

[Welcome](#)  
[Installation](#)  
[Getting Started](#)  
[Settings](#)  
[Basic Functions](#)  
[Formatting](#)  
[Reports](#)  
[Messaging](#)  
[Advanced Functions](#)

Home > Data Manager > The Connection String

## The Connection String

Form Master 2008 allows the user to specify a Connection String to a MS-SQL Database external to the Database running the DotNetNuke® Application. Once a Form is Mapped to a Table the Connection String cannot be changed. If the Connection String needs to change for some reason, it will be necessary to Re-Map the Fields.

**Form Data Table Manager**

Current Mapping    Map Existing Table    Create Table From Form    Create Form From Table

**Connection String:**  
Server=myServerAddress;Database=myDataBase;Uid=myUsername;Pwd=myPassword;

[Save Connection String](#)

**Current Mapping Information**

The Form is not mapped to any table, and is using the default XML Data Store.

See [www.connectionstrings.com](http://www.connectionstrings.com) for a good connection string reference

### See also

[Map Existing Table](#)  
[Table From Form](#)  
[Form From Table](#)  
[The Mapped Form](#)  
[Edit Mappings](#)  
[External Table Mask](#)

[Home](#) > [Data Manager](#) > [Map Existing Table](#)

## Map Existing Table

Form Master 2008 can map Form Fields to an Existing SQL Table.

**Form Data Table Manager**

Current Mapping   
  Map Existing Table   
  Create Table From Form   
  Create Form From Table

**Connection String:**  

  
[Save Connection String](#)

**Data Source Table List:**  

▼

Column	Type	Length	Prec	Scale	Nullable	Form Field	Map	UnMap
ContactID	int	4	10	0	no	Key		
Name	nvarchar	50			no	Name	<a href="#">Map</a>	<a href="#">UnMap</a>
Email	nvarchar	100			no		<a href="#">Map</a>	<a href="#">UnMap</a>
Subject	nvarchar	100			no		<a href="#">Map</a>	<a href="#">UnMap</a>
Message	nvarchar	50			yes		<a href="#">Map</a>	<a href="#">UnMap</a>

[Commit Operation](#)   
 [Cancel](#)

- Connection String: The Connection string to the Data Resource. Leave blank to use the Default System Connection String. If defining a Connection string, it must be set before any Mappings are started since Mappings are based on what Tables exist in the Database.
- Data Source Table List: The List of Tables from the Default DNN or User Specified Database
- Existing Table Fields: The Grid shows the Fields in the Selected Table and allows for Mapping to Form Master Created Fields.
- Form Field: List of Fields Defined in the Form used for Mapping to External Table Fields.

The Grid Columns:

- Column: The name of the Table Column
- Type: The Data Type of the Table Column
- Length: The allowable Length of data for the Column. This is useful primarily for Text type data and does not apply to Integer types or Boolean values. A discussion of SQL Data Types is beyond the scope of this User Guide.
- Prec & Scale: These are for use with Numeric & Decimal Data Types
- Nullable: If a column is not Nullable then a Required Field Validator will need to be included on the Form.
- Form Field: Initially empty, will show Form Field names as they are mapped to the Table.

The Process:

1. Select the "Map Existing Table" Radio Button
2. Optionally Specify a Connection String
3. Select a Table to map to in the "Data Source Table List" dropdown list. Note: the selected table must have an Auto Increment Identity Column
4. Select a Form Field Radio Button from the "Form Fields" list
5. Click on the "Map" link on the Table Column to be mapped to the Form Field. If a mistake is made the UnMap link can be used to remove the mapping.
6. When all Form Fields have been mapped, click the "Commit Operation" button.

See also

[The Connection String](#)  
[Table From Form](#)  
[Form From Table](#)  
[The Mapped Form](#)  
[Edit Mappings](#)  
[External Table Mask](#)

Home > Data Manager > Table From Form

## Table From Form

Form Master 2008 can create an External Table from the Fields defined in the Form.

**Form Data Table Manager**

Current Mapping   
  Map Existing Table   
  Create Table From Form   
  Create Form From Table

**Connection String:**

[Save Connection String](#)

**New Table Name:**

**Create Table From Form**

Form Field	Type	Length	Prec	Scale	Nullable	Ignore
Email	NVarChar	150	18	0	<input type="checkbox"/>	<input type="checkbox"/>
Name	NVarChar	100	18	0	<input type="checkbox"/>	<input type="checkbox"/>
Subject	NVarChar	50	18	0	<input type="checkbox"/>	<input type="checkbox"/>
Message	NText	8	18	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>
HiddenField	NVarChar	50	18	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

[Commit Operation](#)   
 [Cancel](#)

- Connection String: The Connection string to the Data Resource. Leave blank to use the Default System Connection String. If defining a Connection string, it must be set before any Mappings are started since Mappings are based on what Tables exist in the Database
- New Table Name: The name for the new Table which will be created in the Database

The Grid Columns:

- Form Field: This will be the name of the column in the SQL Table. This value is taken from the "Field Name" value in the [Column Options](#) section.
- Type: The Data Type for the SQL Table Field
- Length: The Length of the SQL Field. For Text fields, Char, NChar, VarChar, NVarChar the value can be set to the greatest expected value for the field. In cases where the amount of data is unknown the Text or NText Data Types should be used as will allow more data then can be sent in a browser.
- Prec. & Scale: Precision is the number of digits in a number. Scale: Scale is the number of digits to the right of the decimal point in a number. Precision and Scale only apply to Numeric and Decimal Data Types and will be ignored otherwise
- Nullable: If Checked the SQL Table will allow Null as a Value in the Column
- Ignore: If Checked the Column will not be created in the SQL Table and no Mapping will be created for it. This is useful for Hidden Fields or Conformation type Fields which do not need to be stored.

The Process:

1. Select the "Create Table From Form" Radio Button
2. Enter a Name for the New Table (a name with no spaces or special characters is recommended)
3. Choose the appropriate Data Type and Length for the Form Field
4. Check the Nullable Checkbox if there is no Required Field Validator on the Field
5. Check if the Field should be Ignored
6. Click the "Commit Operation" Button

See also

[The Connection String](#)  
[Map Existing Table](#)  
[Form From Table](#)  
[The Mapped Form](#)  
[Edit Mappings](#)  
[External Table Mask](#)

## Form From Table

Form Master 2008 can create a Form from the Fields in an External Table. Using this function will overwrite any existing form for this Module.

**Form Data Table Manager**

○ Current Mapping ○ Map Existing Table ○ Create Table From Form ○ Create Form From Table

**Connection String:**

Save Connection String

**Data Source Table List:**

**Create Form From Table**

Column	Type	Length	Prec	Scale	Nullable	Field Type	Form Field	Required
ContactID	int	4	10	0	no	TextBox	<input type="text"/>	<input checked="" type="checkbox"/>
Name	nvarchar	50			no	TextBox	<input type="text" value="Name"/>	<input checked="" type="checkbox"/>
Email	nvarchar	100			no	TextBox	<input type="text" value="Email"/>	<input checked="" type="checkbox"/>
Subject	nvarchar	100			no	TextBox	<input type="text" value="Subject"/>	<input checked="" type="checkbox"/>
Message	nvarchar	50			yes	TextBox	<input type="text" value="Message"/>	<input checked="" type="checkbox"/>
SendInfo	bit	1			yes	CheckBox	<input type="checkbox"/>	<input type="checkbox"/>
Country	nchar	2			yes	DropDownList	<input type="text"/>	<input type="checkbox"/>

- Data Source Table List: The List of Tables from the Default DNN or User Specified Database

The Grid Columns:

- Column: The name of the SQL Table Column
- Type: The Data Type of the Table Column
- Length: The allowable Length of data for the Table Column. This is useful primarily for Text type data and does not apply to Integer types or Boolean values and is shown for setting up field validation. A discussion of SQL Data Types is beyond the scope of this User Guide.
- Prec & Scale: These are for use with Numeric & Decimal Data Types, and is shown for setting up field validation
- Nullable: If a column is not Nullable then the Required Checkbox should be checked.
- Form Field: The Form Field name which will be specified in the Field Name textbox in [Common Options](#) (this name must not contain spaces or special characters)

The Process:

1. Select the "Create Form From Table" Radio Button
2. Select the Field Type
3. Enter a Form Field Name or use the supplied Name
4. Check if the Field will have a Required Field Validator added
5. Click the "Commit Operation" Button

The Connection String

[Home](#) > [Data Manager](#) > [The Mapped Form](#)

## The Mapped Form

Once a Form is mapped to a Table Form Master 2008 will show the page below. Once a Form is Mapped all mapping options and Connection String options will be un-available. In a newly mapped form which previously had saved data the option to Import the XML data will be available. Once the data has been imported the "Import Data Stored as XML" button will not be available.

**Form Data Table Manager**

Current Mapping   
  Map Existing Table   
  Create Table From Form   
  Create Form From Table

**Connection String:**

**Current Mapping Information**

The Form Fields are Mapped to Table: [Contacts]

Column	Type	Length	Prec	Scale	Nullable	Form Field
Name	nvarchar	50			No	Name
Email	nvarchar	100			No	Email
Subject	nvarchar	100			No	Subject
Message	nvarchar	50			No	Message
SendInfo	bit	1			No	SendInfo
Country	nchar	2			No	Country

The Mappings can be deleted by using the "Delete Mappings" button. The Mappings can easily be done again which allows for easy changes to the database schema.

To Import data from the default data store click the "Import Data Stored as XML" button.

**Form Data Table Manager**

Current Mapping   
  Map Existing Table   
  Create Table From Form   
  Create Form From Table

**Connection String:**

**Xml Data Import Results**

Table Name: Contacts   
 Total Records: 2   
 Successful Records: 2   
 Failed Records: 0

No Errors Occured During Import of Xml Data

Form Master 2008 will report the results of the import and show any records which failed the import process.

## Form Data Table Manager

Current Mapping  
  Map Existing Table  
  Create Table From Form  
  Create Form From Table

**Connection String:**

Save Connection String

**Xml Data Import Results**

Table Name: Contacts

Total Records: 2    Successful Records: 0    Failed Records: 2

FormDataID	Error Message	
278	Input string was not in a correct format.	<a href="#">Delete</a>
279	Input string was not in a correct format.	<a href="#">Delete</a>

[Continue](#)

The Error returned above indicates that the Data Type in the SQL Table is not correct for the type of Data stored in the XML. In this case the error was caused by trying to import two letter Country Codes into an Integer Data Type.

See also

[The Connection String](#)  
[Map Existing Table](#)  
[Table From Form](#)  
[Form From Table](#)  
[Edit Mappings](#)  
[External Table Mask](#)

Home > Data Manager > Edit Mappings

## Edit Mappings

Once a Form has been Mapped to a SQL Table, the mappings can be edited. This may become necessary for example if a new field is added to the SQL Table and a new Form Field is created to collect the new data. In cases like this either the new SQL Table Field will need to allow Nulls or the new column must be populated with default values.

**Form Data Table Manager**

○ Current Mapping ○ Map Existing Table ○ Create Table From Form ○ Create Form From Table

**Connection String:**

Save Connection String

**Data Source Table List:**

**Map Existing Table**

Column	Type	Length	Prec	Scale	Nullable	Form Field		
ContactID	int	4	10	0	no	Key	<input type="button" value="Map"/>	<input type="button" value="UnMap"/>
Name	nvarchar	50			no	Name	<input type="button" value="Map"/>	<input type="button" value="UnMap"/>
Email	nvarchar	100			no	Email	<input type="button" value="Map"/>	<input type="button" value="UnMap"/>
Subject	nvarchar	100			no	Subject	<input type="button" value="Map"/>	<input type="button" value="UnMap"/>
Message	nvarchar	50			yes	Message	<input type="button" value="Map"/>	<input type="button" value="UnMap"/>
SendInfo	bit	1			yes	SendInfo	<input type="button" value="Map"/>	<input type="button" value="UnMap"/>
Country	nchar	2			yes	Country	<input type="button" value="Map"/>	<input type="button" value="UnMap"/>

**Existing Table Fields:**

**Form Fields:**

- Email
- Name
- Subject
- Message
- HiddenField
- SendInfo
- Country

### See also

[The Connection String](#)  
[Map Existing Table](#)  
[Table From Form](#)  
[Form From Table](#)  
[The Mapped Form](#)  
[External Table Mask](#)

## External Table Mask

For Security Reasons Form Master 2008 creates an External Table Mask List in Host > Lists which is used to Mask Tables in the List from the User during Table Mapping Operations in the Default Database. Only the Host Role can manage this list which is partially shown below.

<b>List Details</b>	
<b>List Name:</b>	FormMasterExtTableMask
<b>Total:</b>	70 entries
<a href="#">+ Add Entry</a> <a href="#">X Delete List</a>	
<b>Text</b> <b>Value</b>	
<a href="#">X</a> Affiliates	Affiliates
<a href="#">X</a> AnonymousUsers	AnonymousUsers
<a href="#">X</a> aspnet_Applications	aspnet_Applications
<a href="#">X</a> aspnet_Membership	aspnet_Membership
<a href="#">X</a> aspnet_Profile	aspnet_Profile
<a href="#">X</a> aspnet_Roles	aspnet_Roles
<a href="#">X</a> aspnet_SchemaVersions	aspnet_SchemaVersions
<a href="#">X</a> aspnet_Users	aspnet_Users
<a href="#">X</a> aspnet_UsersInRoles	aspnet_UsersInRoles
<a href="#">X</a> Assemblies	Assemblies
<a href="#">X</a> Authentication	Authentication
<a href="#">X</a> Banners	Banners
<a href="#">X</a> C5SLLC_Form	C5SLLC_Form
<a href="#">X</a> C5SLLC_FormData	C5SLLC_FormData
<a href="#">X</a> C5SLLC_FormFieldRules	C5SLLC_FormFieldRules
<a href="#">X</a> C5SLLC_FormFileData	C5SLLC_FormFileData
<a href="#">X</a> C5SLLC_FormMapping	C5SLLC_FormMapping
<a href="#">X</a> Classification	Classification
<a href="#">X</a> DesktopModules	DesktopModules
<a href="#">X</a> EventLog	EventLog
<a href="#">X</a> EventLogConfig	EventLogConfig
<a href="#">X</a> EventLogTypes	EventLogTypes
<a href="#">X</a> EventQueue	EventQueue
<a href="#">X</a> Files	Files
<a href="#">X</a> FolderPermission	FolderPermission
<a href="#">X</a> Folders	Folders
<a href="#">X</a> HostSettings	HostSettings
<a href="#">X</a> HtmlText	HtmlText
<a href="#">X</a> Lists	Lists
<a href="#">X</a> ModuleControls	ModuleControls
<a href="#">X</a> ModuleDefinitions	ModuleDefinitions
<a href="#">X</a> ModulePermission	ModulePermission

### See also

[The Connection String](#)  
[Map Existing Table](#)  
[Table From Form](#)  
[Form From Table](#)  
[The Mapped Form](#)  
[Edit Mappings](#)

Home > Formatting

## Formatting

Topics in this section



Form Formatting



Email Formatting

Add Content

See also

Welcome  
Installation  
Getting Started  
Settings  
Basic Functions  
Data Manager  
Reports  
Messaging  
Advanced Functions

[Home](#) > [Formatting](#) > [Form Formatting](#)

## Form Formatting

Formatting the form provides visual cues which draw attention to different sections of the form, this improves the User Interface by helping the user perform the task more efficiently, and helps deliver a better user experience..

Applying formatting to Form Master is done using Cascading Style Sheets (CSS), and applying Class Name Rules to elements of the form. The logical place for the Style Rules is in Portal.css since this Stylesheet is specific to each Portal, but the Rules can be in any Stylesheet which get loaded with the Form.

Class names can be applied to the elements below:

- Outer Table
- Inner Table
- Individual Table Cell within the Inner Table
- Labels for each individual fields in the form
- Individual fields within the form

The presentations below will show where and how the Class Names can be applied, and finally, the Class Name definitions.

### [Formatting the Form Presentation](#)

#### **The Class Definitions**

```
.mainReport
{
    border: black 1px solid;
    border-left: black 1px solid;
    width: 600px;
    border-collapse: collapse;
    background:ivory;
}
.frmReport
{
    border: black 1px solid;
    width: 500px;
    border-collapse: collapse;
}
.frmPadding
{
    padding:3px 3px 3px 3px;
}
.addressInfo
{
    background:#eeeeee;
    padding:3px 0px 6px 6px;
}
.frmSeparator
{
    border-bottom:1px black solid;
    height:10px;
}
.widgetHeading
{
    background:brown;
    padding:3px 3px 3px 8px;
}
.chkLstShip
{
    background:wheat;
    width:250px;
    padding:3px;
}
.chkLstMake
{
    background:silver;
    width:250px;
    padding:3px;
```

Formatting • [Email](#)

See also

[Email Formatting](#)

[Home](#) > [Formatting](#) > [Email Formatting](#)

## Email Formatting

### Topics in this section



[XSL Templates](#)



[Tokenized Templates](#)



[Custom Template](#)

Email can be sent in either HTML or Text Formatting. When sent in HTML format Form Master 2008 sends both the HTML and Text in a Multipart Alternative Email Message (Auto Responder messages).

### See also

[Form Formatting](#)

[Home](#) > [Formatting](#) > [Email Formatting](#) > [XSL Templates](#)

## XSL Templates

### XSL Email Templates

Form Master offers two Template styles for formatting the email message which was previously covered in the Other Settings topic. This section will show how the XSL templates work.

Customizing the XSL templates will not be shown as this is beyond the scope of this User Guide

[XSL Email Templates Presentation](#)

### See also

[Tokenized Templates](#)  
[Custom Template](#)

[Home](#) > [Formatting](#) > [Email Formatting](#) > [Tokenized Templates](#)

## Tokenized Templates

Form Master offers two Template styles for formatting the email message which was previously covered in the Other Settings topic.

This section will show how the Tokenized templates work and how to use them to modify the body of the email sent from Form Master.

[Tokenized Email Templates Presentation](#)

### See also

[XSL Templates](#)  
[Custom Template](#)

Home > Formatting > Email Formatting > Custom Templates

## Custom Template

## Custom Template

By using Field **[TOKENS]** and the Custom Template, it is possible to completely customize the email which is sent to the email addresses in the SendTo: field of the Settings > **Form Settings** page

**Email Custom Template**

Basic Text Box  Rich Text Editor

[EMAIL]  
[FULLNAME]  
[SUBJECT]  
[MESSAGE]

**Code 5 Systems**

---

A User has submitted a question on the **Contact Form**. Please find the Information listed below.

Email:	[EMAIL]
Name:	[FULLNAME]
Subject:	[SUBJECT]
Message:	[MESSAGE]

- The body of the message can contain any valid HTML
  - To Insert the user supplied data from the form just use any of the previously defined Field [TOKENS] from the Defined Tokens column on the right.
  - When the Form is submitted, Form Master will replace the [TOKENS] such as Administrator above with the information provided by the user.

#### See also

## XSL Templates

---

Home > Reports

## Reports

Topics in this section



[Viewing Reports](#)



[Downloading Reports](#)

Add Content

See also

Welcome  
Installation  
Getting Started  
Settings  
Basic Functions  
Data Manager  
Formatting  
Messaging  
Advanced Functions

[Home](#) > [Reports](#) > [Viewing Reports](#)

## Viewing Reports

### Reports

When "Save Data to Database" is checked in the "Form Options", reports will be available for viewing, editing, downloading, and of course deleting. Reporting and deleting is done based on a user specified date range, the default is a range covering the past 30 days from the current date.

### Viewing Reports

If files have been uploaded, links are available in the form to download the files. Each record can be edited, and even uploaded files can be replaced or added to records where files have not been uploaded.

Records can be deleted individually or by date range. Any uploaded files associated with any deleted record are deleted from the database or from the file system.

[Viewing Reports Presentation](#)

### See also

[Downloading Reports](#)

---

[Home](#) > [Reports](#) > [Downloading Reports](#)

## Downloading Reports

### Downloading Reports

Download of data is available in four formats:

- Excel 2007: Use this link if you are using Excel 2007. Excel will complain about the format since it no longer supports HTML Table formatting. Clicking Yes and Excel will open it in 2003 Compatibility Mode.
- Excel 2002 & 2003: Use this option for Excel versions 2002 and 2003. This method downloads Html to Excel. After the file is saved to disk, it can be opened in Excel and saved in Excel Native format using the "Save As" menu item.
- CSV: Comma Separated Values: This is a standard format for sharing information between disparate databases. The fields are separated by commas using double quotes as field delimiters.
- Tab Delimited: Again another standard format. Fields are separated by Tabs with no field delimiters.

[Downloading Reports Presentation](#)

### See also

[Viewing Reports](#)

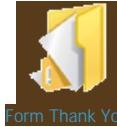
[Home](#) > [Messaging](#)

## Messaging

Topics in this section



Form Heading



Form Thank You



Auto Responder

See also

[Welcome](#)  
[Installation](#)  
[Getting Started](#)  
[Settings](#)  
[Basic Functions](#)  
[Data Manager](#)  
[Formatting](#)  
[Reports](#)  
[Advanced Functions](#)

Home > Messaging > Form Heading

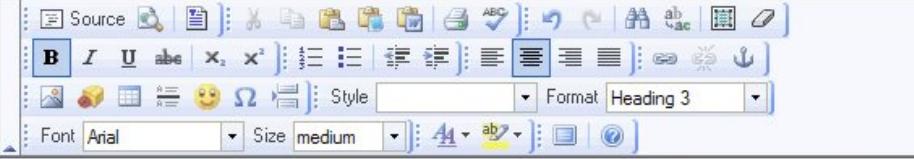
## Form Heading

The Form Heading allow for the insertion of Html content at the top of the form, in the first row of the "Outer Table" element.

The default text used to provide instructions when the Module is first placed on the page, should either be deleted or replaced by custom content for the current form

**Form Heading Message**

Basic Text Box  Rich Text Editor



**Please Complete the Form Below for a Quick response to your needs.**

Show custom editor options | Refresh Editor

Update!

### See also

[Form Thank You](#)  
[Auto Responder](#)

Home > Messaging > Form Thank You

## Form Thank You

The Form Thank You allows for the messaging after the form has been submitted. This message will be shown based on the setting of the "Redirection After Submit" in the "Other Settings" page. The options which will show the Form Thank You message are "None" and "File"

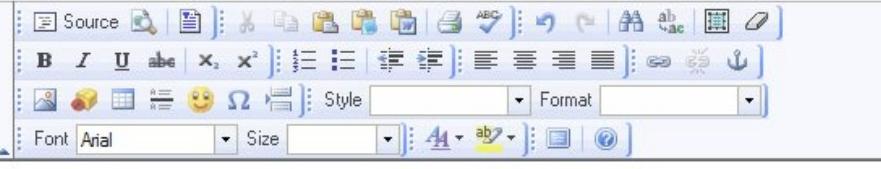
For the "File" option, it is important to provide the user with instructions to click the link provided if the file does not start to download automatically

When Field Tokens are defined in the fields they will show in the "Defined Tokens" area and can be used in the body of the Thank You message. These Tokens will be replaced with the User Entered data when the form is submitted.

**Thank You Message**

Note: Form Must be set to "Save Data to Database" in order to use Tokens

Basic Text Box  Rich Text Editor



[FIRSTNAME], Thank you very much for your interest in our products. You will be hearing from us shortly.  
My Company

**Defined Tokens**

[FIRSTNAME]
[LASTNAME]

### See also

[Form Heading](#)  
[Auto Responder](#)

[Home](#) > [Messaging](#) > [Auto Responder](#)

## Auto Responder

Form Master allows for an Automated Response to be sent to the user when the form is submitted, by using the Auto Responder options on the Other Settings page.

Auto Response Email Body text can contain any HTML suitable for email messaging. Any image URL need to be absolute URL in order for the image to show in the user's email client. The email body is designed using the Messaging > Auto Responder menu option.

The Auto Responder is not dependant on the state of any other options in Form Master, and can be sent in any configuration

When Field Tokens are defined in the fields they will show in the "Defined Tokens" area and can be used in the body of the Auto Responder message. These Tokens will be replaced with the User Entered data when the message is sent.

The screenshot shows the 'Auto Responder Email' editor. On the left is a rich text editor toolbar with various icons for text, images, and styling. Below the toolbar is a preview area containing the following content:

**Code 5 Systems**

Dear [FIRSTNAME] [LASTNAME],

Thank you for your interest in our Products and Services.

Attached please find a brochure detailing our full line of software products.

Sincerely,  
The Code 5 Systems Team

On the right side of the editor, there is a sidebar titled 'Defined Tokens' which lists the tokens [FIRSTNAME] and [LASTNAME].

See also

[Form Heading](#)  
[Form Thank You](#)

[Home](#) > [Advanced Functions](#)

## Advanced Functions

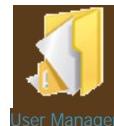
Topics in this section



Silent Post



Field Rules



User Manager

See also

[Welcome](#)  
[Installation](#)  
[Getting Started](#)  
[Settings](#)  
[Basic Functions](#)  
[Data Manager](#)  
[Formatting](#)  
[Reports](#)  
[Messaging](#)

## Silent Post

The Silent Post feature is useful for sending information to outside Partners, Vendors, or another process in the same Website.

A classic example of using the Silent Post feature is when using an outside Vendor to manage a Continuity Marketing Program, or CRM management. These programs are offered by many Vendors, the most notable are Constant Contact®, AWeber®, Vertical Response®, and SalesForce®.

In these services it is common for the vendor to supply code which shows a Form where an end user can sign up for a Mailing List, or some other service which the Vendor offers. The Vendors Client is instructed to place the code in their Web Page. This works fine for plain HTML pages which can have multiple Form Tags in the page. For DotNetNuke this becomes a problem, as it is difficult in ASP.NET and DotNetNuke to include a FORM tag in the page.

Using the Silent Post Feature a Vendor Form can be Mimicked with the appropriate end User Visible Fields and the required Vendor Hidden fields. When the Form Master 2008 Form is submitted the Vendor form is Created and Posted in the Background to the URL supplied by the Vendor in their Forms Tag "Action" attribute. The series of events is shown below

1. End User enters information into the Form Master form and Submits
2. Data is collected and stored according to the options specified
3. The Specified Silent Post Fields are used to create an HTTP Request to the Vendor URL
4. The end user does not know anything about the Silent Post activity, Form Master handles the End User as specified in Settings > Other Settings as usual.

For an example of setting up a form to work with SalesForce® please see this [Salesforce® Case Creation Form](#)

### See also

## Field Rules

Form Master 2008 supports conditional field display rules. For example in a Form where we have a DropDownList of Regions populated with US States and Canadian Provinces, if a User is not from one of these countries, we should provide an option in the DropDownList for "Other", and a TextBox where the user can fill in a Region. Field Rules will allow us to show the Other Region TextBox if the value of the DropDownList is "other" and hide it when it is not. Below is the Field Rules page.

**Form Field Rules**

<b>When Source Field:</b> <input type="button" value="&lt;&lt; Select &gt;&gt;"/>	<b>Contains/Is:</b> <input type="text"/>
<b>Show Target Field(s):</b>	
<input type="checkbox"/> Email <input type="checkbox"/> Name <input type="checkbox"/> Subject <input type="checkbox"/> Message <input type="checkbox"/> SendInfo <input type="checkbox"/> Country <input type="checkbox"/> Region <input type="checkbox"/> OtherRegion	
<b>Add Rule</b>	

<b>Existing Rules</b>			
<b>Source Field</b>	<b>Contains/Is</b>	<b>Show Fields(s)</b>	
Region	other	OtherRegion	<a href="#">Delete</a>

- When Source Field: The Field for which the Value will cause other Fields to be shown or hidden. This Dropdown List contains the defined form Fields excluding incompatible types such as Hidden, File Upload, Button, and Label Fields.
- Contains/Is: For Checkboxes the default show value is "Checked". All other field types, the show value will be Contains. A Partial value will be matched for example:
  - A DropDownList of Regions contains "Other Region" and a TextBox should be shown when the user selects that value. If "Other" is entered into the Contains TextBox it will match the DropDownList value.
  - The Contains TextBox can also contain a wildcard of \*.\* which will match any value Target Fields. This works for any type of Source Field.
- Show Target Fields(s): Select One or More Fields to Show when the Rule is True
- Existing Rules: The grid shows existing rules for this form.

The Grid Columns:

- Source Field: The Form Field which Triggers the Rule
- Contains/Is: The Value which the Source Field must Contain, or for Checkboxes be Checked to Trigger the Rule
- The Grid Columns: The list of Fields which will be shown when the Rule Condition is True.

The Process:

- Select a Source Field from the "When Source Field:"
- Enter a value which the Source Field must contain to Trigger the Rule. For Source Field of type Checkbox the value defaults to Checked
- Select one or more fields to be the target of the Rule in the "Show Target Field(s)" Checkbox List
- Click the "Add Rule" button

Field Rule In Action: Not Triggered

Name:\*

Email:\*

Region: << Select >>

Country: << Select >>

Subject:\*

Send Information

Message:\*

Submit

Field Rule In Action: Triggered

Name:\*

Email:\*

Region: Other

Country: << Select >>

Other Region:

Subject:\*

Send Information

Message:\*

Submit

See also

[Silent Post](#)  
[User Manager](#)

Home > Advanced Functions > User Manager

## User Manager

Form Master 2008 can Register a User and add the user to any number of Roles when the Form is Submitted. If a user is already logged in the Registration will be ignored, but the user will still be added to any Roles which they are not already members of. Below is the User Manager page which is used to manage this feature.

Note: User Manager is available for Super Users and users in the Administrators Role only.

**User Manager**

 **Register User**

 **User Name Field**  
 

 **Password Field**  
 

 **First Name Field**  
 

 **Last Name Field**  
 

 **Email Field**  
 

 **User Roles**

**Charter**    **Maximizers**    **Premier**    **Subscribers**

**Developers**    **Performers**    **Strivers**    **Untouchables**

[Update](#)

- Register User: Check to Register the User when the Form is Submitted. Fields to be used for the UserName, Password, First Name, Last Name, and Email must be defined. If user is logged in this setting is ignored.
- User Name Field: Field to be used as the User Name when Registering a User
- Password Field: Field to be used as the Password when Registering a User
- First Name Field: Field to be used as the First Name when Registering a User
- Last Name Field: Field to be used as the Last Name when Registering a User
- Email Field: Field to be used as the Email when Registering a User
- User Roles: Select The Roles the User should be Added to when Submitting the Form. Note: User Must be logged in or being Registered during submission.

### See also

[Silent Post  
Field Rules](#)

# Index

## A

About Form Master 2008 7  
Adding Fields 22  
Advanced 72  
Advanced: Field Rules 74  
Advanced: Silent Post 73  
Advanced: User Manager 76

Fields: Listbox 28  
Fields: Radiobutton List 30  
Fields: Submit Button 33  
Fields: Textbox 25  
File Management 48  
Form Elements 9  
Form Layout 19  
Formatting 58  
Formatting: The Form 59

## D

Data Manager 49  
Data: Connection Strings 50  
Data: Edit Mappings 56  
Data: External Table Mask 57  
Data: Form From Table 53  
Data: Map Existing Table 51  
Data: Mapped Form 54  
Data: Table From Form 52

## E

Email Formatting 61  
Email Formatting: Custom 64  
Email Formatting: Tokenized 63  
Email Formatting: XSL 62

## G

Getting Started 6

Installing Form Master 2008 5

## L

Localization 47

## M

Menu Items 10  
Messaging 68  
Messaging: Auto Responder 71  
Messaging: Form Heading 69  
Messaging: Form Thank You 70

## F

Field Options 36  
Field Options: Auto Fill 37  
Field Options: Field Query 39  
Field Options: Send Value By 40  
Field Options: Textbox 38  
Fields: Checkbox 26  
Fields: Checkbox List 29  
Fields: Common Options 23  
Fields: Dropdown 27  
Fields: File Upload 32  
Fields: Hidden Field 34  
Fields: Label Control 31

## R

Reports 65  
Reports: Downloading 67  
Reports: Viewing 66

## S

Settings 11  
Support 2

## System Requirements 4

### T

#### Tab Index 21

### V

Validation 41  
Validation: Checkbox List 45  
Validation: Compare 44  
Validation: Regular Expression 43  
Validation: Required Field 42  
Validation: Summary 46

### W

#### Workflow 8

Form Master 2008  
Copyright © 2005 - 2008  
All Rights Reserved